



User Manual

Version 1.0.0 Aug. 2022

Node-RED

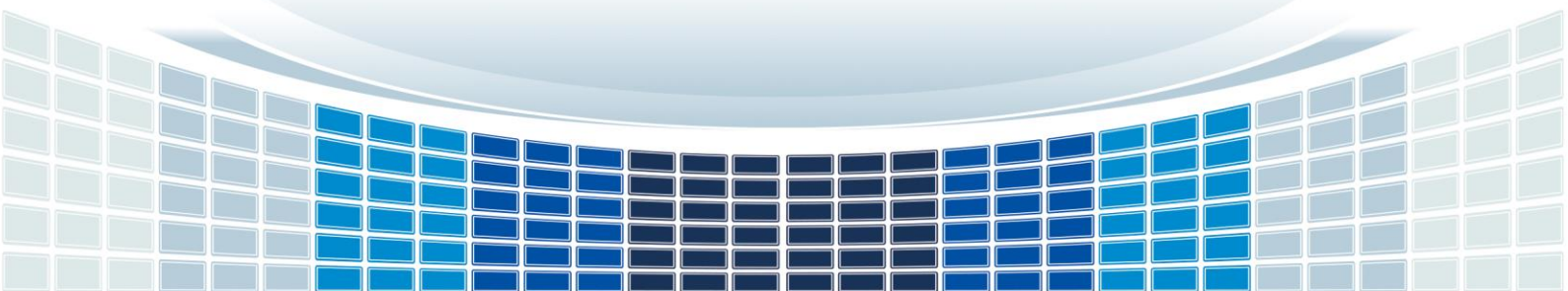
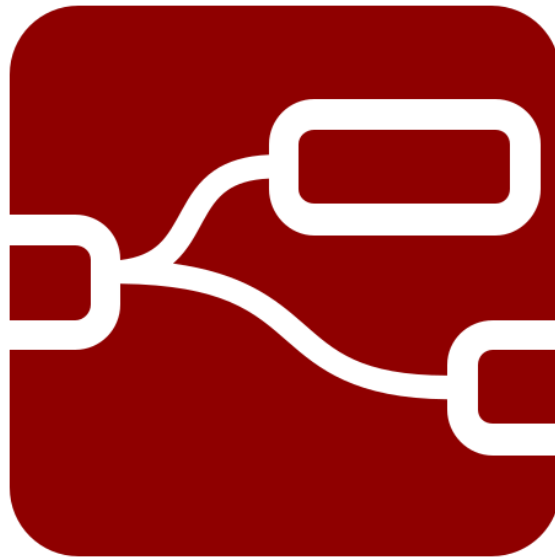


Table of Contents

1. Introduction	4
1.1 Features.....	4
1.2 License.....	5
2. Software Installation	6
3. Node-RED	7
3.1 Editor.....	7
3.1.1 Access the editor	7
3.1.2 Editor window	7
3.2 Workspace	8
3.2.1 Flows	9
3.2.2 Nodes	10
3.2.3 Wires	10
3.2.4 Importing and Exporting Flows	11
3.3 Palette.....	13
3.3.1 Palette Manager	13
3.4 Sidebar.....	15
4. Demo	16
4.1 Demo1 - RTU IO and GPS positioning.....	16
4.2 Demo2 - MQTT publish & subscribe	20
4.3 Demo3 - Read/Write message by serial port	24
4.4 Demo4 - Capture image via webcam	28
4.5 Demo5 - Machine Learning with Google's Teachable Machine.....	30
Appendix A. Revision History	36

Important Information

Warranty

All products manufactured by ICP DAS are under warranty regarding defective materials for a period of one year, beginning from the date of delivery to the original purchaser.

Warning

ICP DAS assumes no liability for any damage resulting from the use of this product. ICP DAS reserves the right to change this manual at any time without notice. The information furnished by ICP DAS is believed to be accurate and reliable. However, no responsibility is assumed by ICP DAS for its use, not for any infringements of patents or other rights of third parties resulting from its use.

Copyright

Copyright © 2022 by ICP DAS Co., Ltd. All rights are reserved.

Trademark

Names are used for identification purpose only and may be registered trademarks of their respective companies.

Contact us

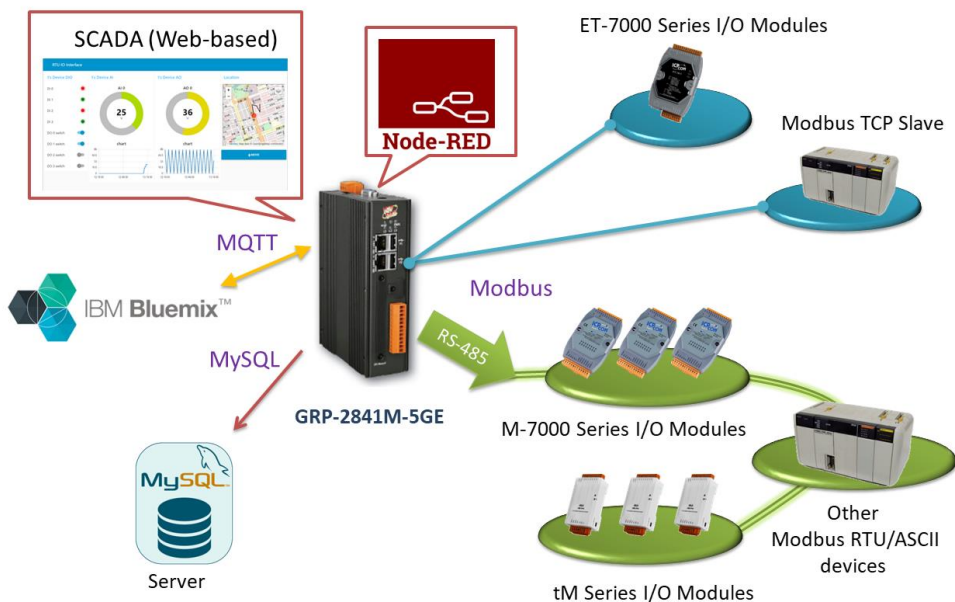
If you have any problem, please feel free to contact us. You can count on us for quick response.

Email: service@icpdas.com

1. Introduction

GRP-2841M / GRP-2841M-5GE supports Node-RED software. Node-RED is a flow-based development tool for visual programming developed originally by IBM for wiring together hardware devices, APIs and online services as part of the Internet of Things.

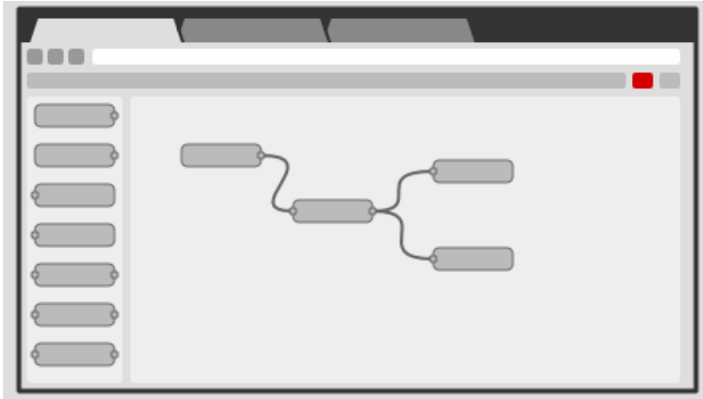
Node-RED provides a web browser-based flow editor, which can be used to create JavaScript functions. Elements of applications can be saved or shared for re-use. The runtime is built on Node.js. The flows created in Node-RED are stored using JSON.



1.1 Features

◆ Browser-based flow editing

Node-RED provides a browser-based flow editor that makes it easy to wire together flows using the wide range of nodes in the palette. Flows can be then deployed to the runtime in a single-click.



◆ Built on Node.js

The light-weight runtime is built on Node.js, taking full advantage of its event-driven, non-blocking model.

◆ Social Development

The flows created in Node-RED are stored using JSON which can be easily imported and exported for sharing with others.

An online flow library allows you to share your best flows with the world.

1.2 License

Node-RED is licensed under the Apache License 2.0

Name	Node-RED
Author	OpenJS Foundation & Contributors
URL	https://nodered.org .

For more information, please refer to the official website: <https://nodered.org/>

2. Software Installation

GRP-2841M has no built-in Node-RED by default and needs to be installed separately by the user. Users can refer to the “Update firmware by pen drive in How to update firmware.pdf file” for installation. Please download the software package (Node-RED) from:
<https://www.icpdas.com/en/download/index.php?model=GRP-2841M-5GE>

file content ➔

Node-Red

- bin (install file)
- updateFile.tarc
- doc (document)
- Node-Red_usermanual_en.pdf (this file)
- demo

Warning:

- 1.It takes about 7 minutes to install this software, please wait patiently for the L1, L2, L3 indicators to light up continuously.



L1 on: find update file

L2 on: updating

L3 on: update completed

- 2.If the installation is successful, the user can access it with the url:
`http://<ip-address>:1880`

3. Node-RED

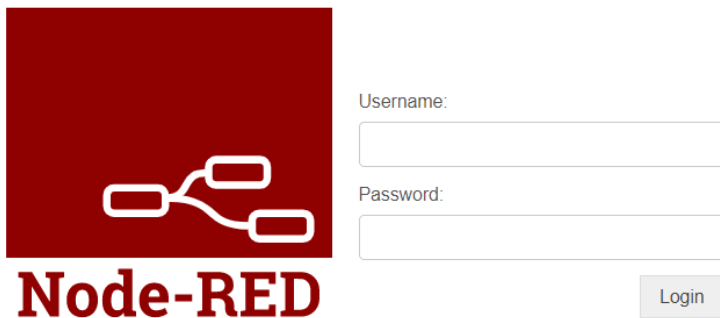
3.1 Editor

3.1.1 Access the editor

With GRP-2841M(-5GE) running, open the editor in a web browser.
You will need to use the ip address of the GRP-2841M(-5GE) to access it with the url: <http://<ip-address>:1880>.

Default username: admin

Default password: admin_GRP

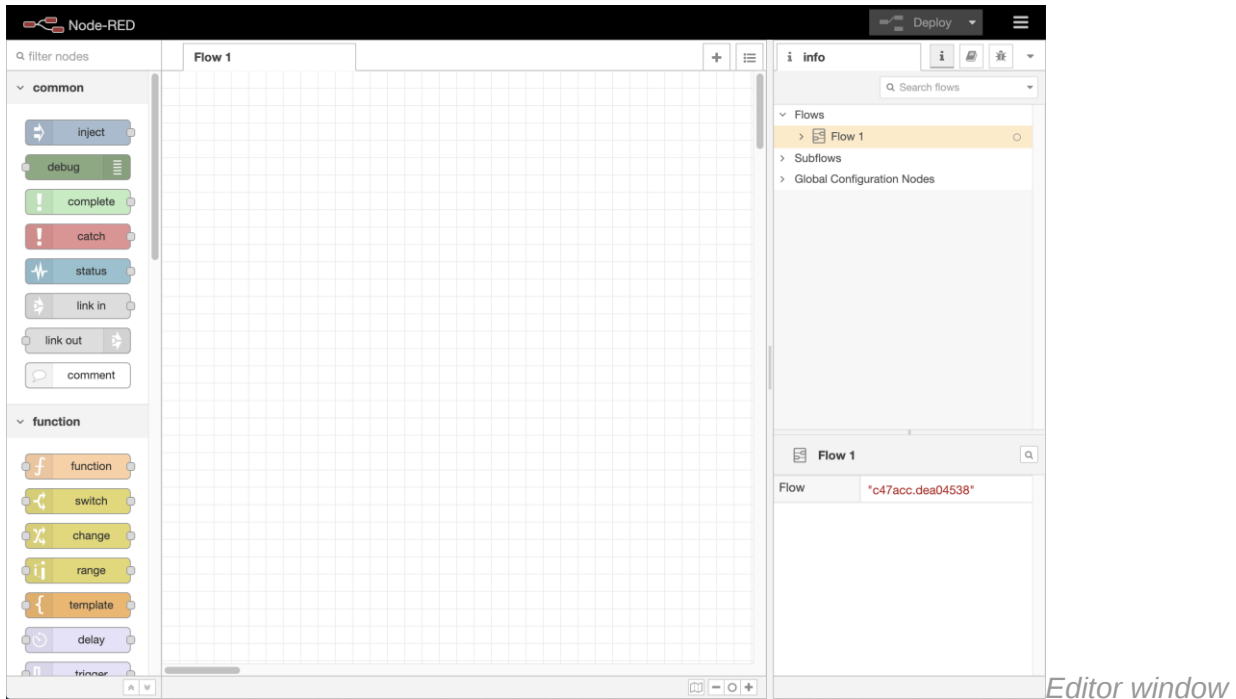


* The user can change the password on the GRP-2841M webpage (System ->Password).

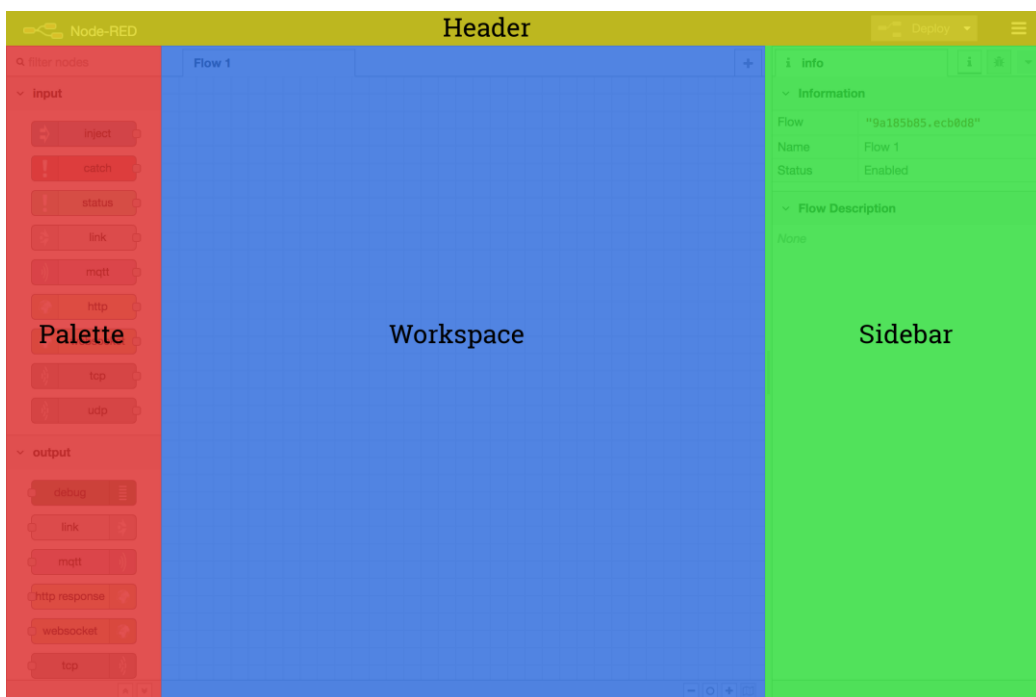
3.1.2 Editor window

The editor window consists of four components:

- The **header** at the top, containing the deploy button, main menu, and, if user authentication is enabled, the user menu.
- The **palette** on the left, containing the nodes available to use.
- The **main** workspace in the middle, where flows are created.
- The **sidebar** on the right.



Editor window



Editor components

3.2 Workspace

The main workspace is where flows are developed by dragging nodes from the palette and wiring them together.

The workspace has a row of tabs along the top; one for each flow and any sub flows that have been opened.

3.2.1 Flows

A flow is represented as a tab within the editor workspace and is the main way to organize nodes.



Each flow can have a name, and description that is displayed in the Information sidebar. All of the nodes in a flow can access the same flow-scoped context.

➤ Adding a flow

To add a new flow, either click the  button in the top bar, or double-click on free space in the tab bar.

➤ Editing flow properties

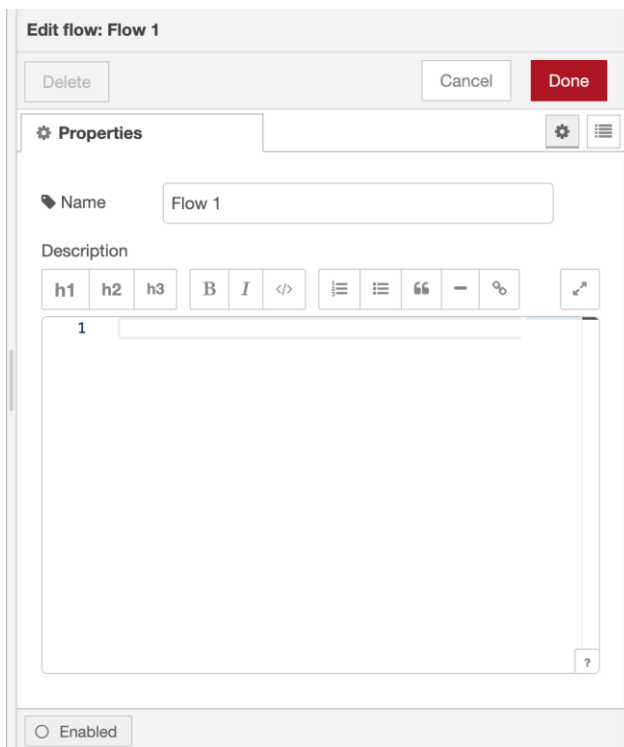
To edit a flow's properties, double-click on its tab in the top bar. This will open the Flow Properties dialog.



: Properties - set the flow's name and description. The description can use Markdown syntax for formatting and will appear in the Information sidebar.



: Environment Variables - properties that are exposed as environment variables within the flow. Since Node-RED 2.1



Flow properties editor

➤ Hiding or showing a flow

A flow can be hidden by clicking on the  on the tab.

When hidden, the Information sidebar will show an  icon next to it. Clicking that icon will show the flow again.

➤ Deleting a flow

To delete a flow, click the 'Delete' button in the Flow Properties dialog.

3.2.2 Nodes

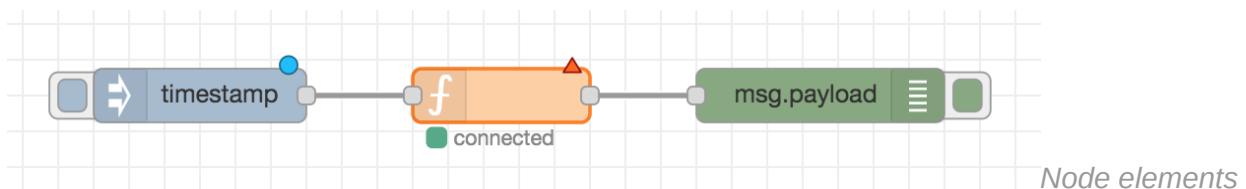
Nodes can be added to the workspace by either:

dragging them from the palette,

using the quick-add dialog (opened by holding the **Ctrl** when clicking on the workspace),
or importing from the library or clipboard.

Nodes are joined together by wires via their ports. A node can have at most one input port and many output ports. A port may have a label that is displayed when the mouse hovers over it. A node may specify labels, for example, the Switch node shows the rule that matches the port. The labels can also be customized in the node edit dialog.

Some nodes display a status message and icon below the node. This is used to indicate the runtime state of the node - for example, the MQTT nodes indicate if they are currently connected or not.

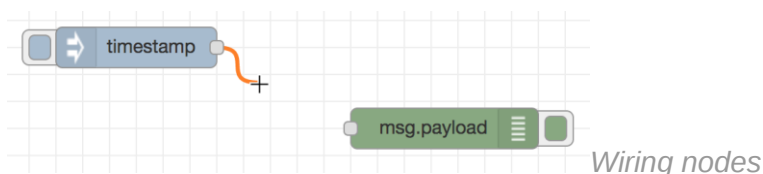


If a node has any undeployed changes, it displays a blue circle above it. If there are errors with its configuration, it displays a red triangle.

Some nodes include a button on either its left or right edge. These allow some interaction with the node from within the editor. The Inject and Debug nodes are the only core nodes that have buttons.

3.2.3 Wires

Nodes are wired together by pressing the left-mouse button on a node's port, dragging to the destination node and releasing the mouse button.



3.2.4 Importing and Exporting Flows

Flows can be imported and exported from the editor using their JSON format, making it very easy to share flows with others.

➤ Importing flows

The Import dialog can be used to import a flow by the following methods:

- pasting in the flow JSON directly,
- uploading a flow JSON file,
- browsing the local flow library,
- browsing the example flows provided by installed nodes.

In all cases, the dialog offers the option to import the nodes into the current flow, or to create a new flow for them.

The screenshot shows the 'Import nodes' dialog box. It has a title bar 'Import nodes'. The main area is divided into two sections: 'Clipboard' and 'Library'. The 'Clipboard' section has a text input field with the placeholder text 'Paste flow json or' and a button 'select a file to import'. The 'Library' section has a list of items. At the bottom, there's an 'Import to' section with two buttons: 'current flow' and 'new flow'. At the very bottom right, there are 'Cancel' and 'Import' buttons.

Import dialog

➤ Exporting flows

The Export dialog can be used to copy flow json out of the editor by the following methods:

- copying the JSON to the system clipboard,
- downloading the JSON as a file,
- saving it to the local flow library.

It can export either the selected nodes, the current flow (including its tab node) or the complete flow configuration.

It offers the option to export compact or formatted JSON. The compact option generates a single line of JSON with no whitespace. The formatted JSON option is formatted over multiple lines with full indentation - which can be easier to read.

Export nodes

Export

selected nodes

current flow

all flows

Clipboard

Local

My Team Library

Export nodes

timestamp

set msg.payload

Change Node

JSON

Cancel

Download

Copy to clipboard

Export Flows dialog

3.3 Palette

The palette contains all of the nodes that are installed and available to use.

They are organized into a number of categories, with inputs, outputs and functions at the top. If there are any subflows, they appear in a category at the top of the palette.

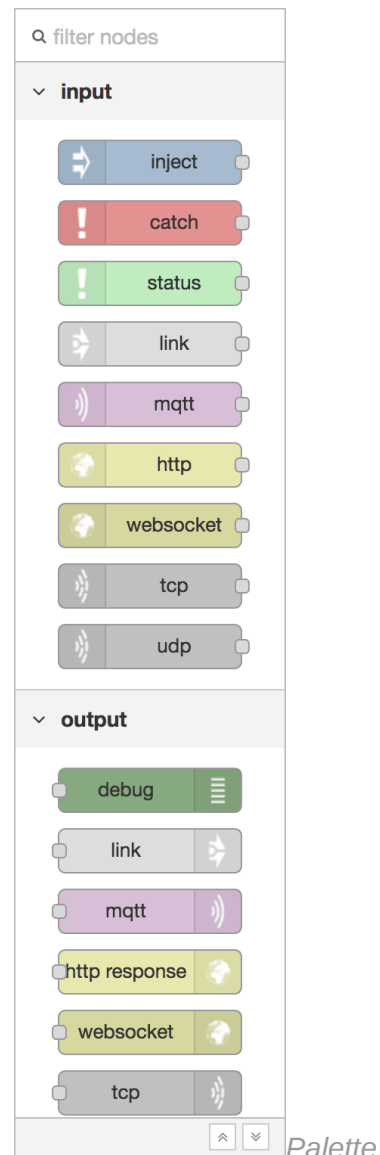
Categories can be expanded or collapsed by clicking its header.

The  and  buttons at the bottom of the palette can be used to collapse or expand all categories. Above the palette is an input that can be used to filter the list of nodes.

The entire palette can be hidden by clicking the palette toggle that is shown when the mouse is over the palette.

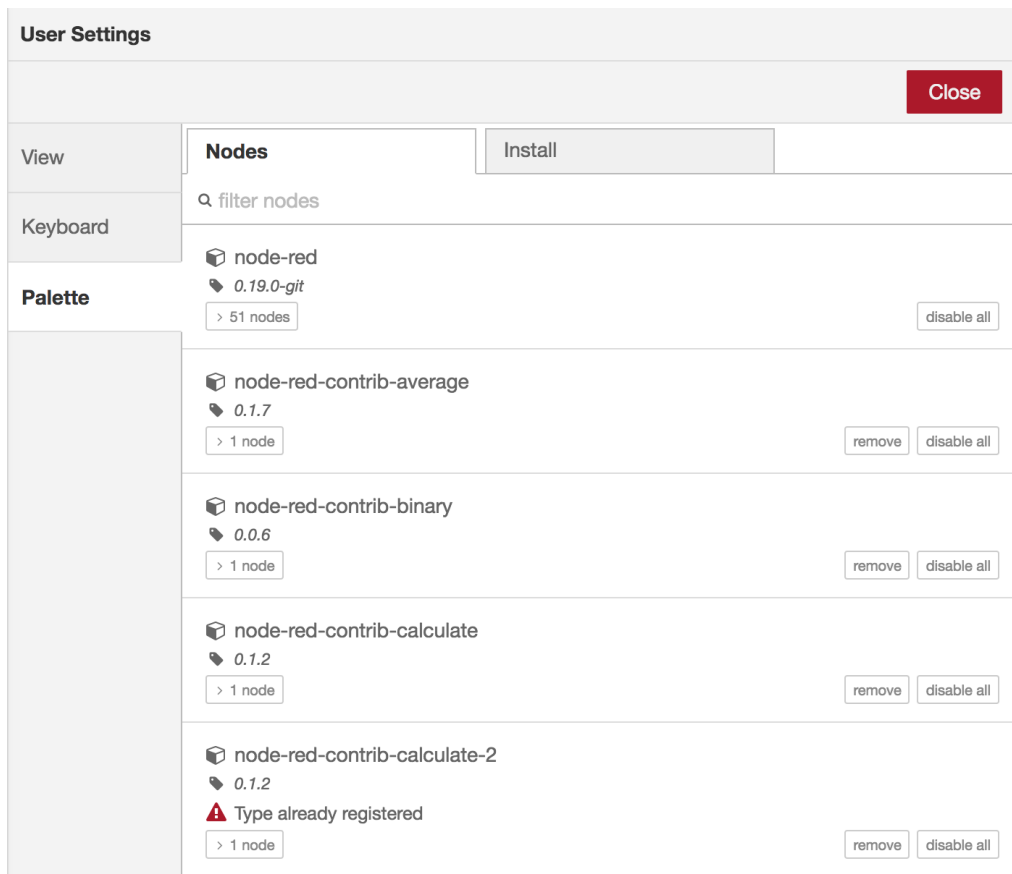


Palette toggle



3.3.1 Palette Manager

The Palette Manager can be used to install new nodes into the palette. It can be accessed under the Palette tab of the User Settings dialog.



Palette Manager

The Palette Manager has two tabs:

- Nodes lists the modules that are currently installed in the runtime
- Install lists the available modules that can be installed

➤ Managing nodes

Each entry in the Nodes list shows the name and version of the module along with a list of the individual node types the module provides.

Options are provided to remove, disable or upgrade each module. If a node is currently in use within the flow, it cannot be removed or disabled.

➤ Installing nodes

The Install tab can be used to search for available modules and install them.

To search for a module, enter its name in the search bar. The search results show the details of the modules, including when it was last updated and a link to its documentation. It can be installed by clicking the 'install' button.

3.4 Sidebar

The sidebar contains panels that provide a number of useful tools within the editor.

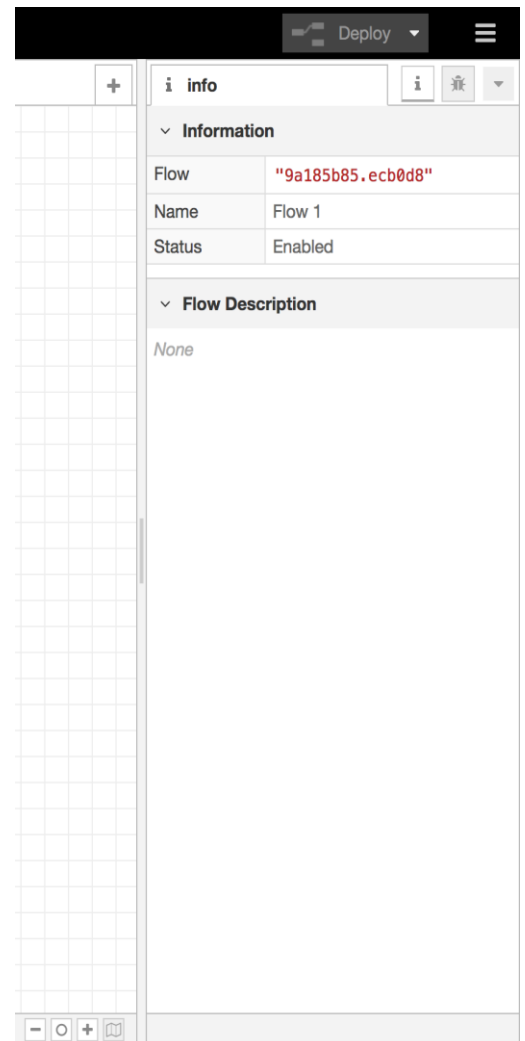
- [Information](#) - view information about nodes and their help
- [Debug](#) - view messages passed to Debug nodes
- [Configuration Nodes](#) - manage configuration nodes
- [Context data](#) - view the contents of context

Some nodes contribute their own sidebar panels, such as node-red-dashboard.

The panels are opened by clicking their icon in the header of the sidebar, or by selecting them in the drop-down list shown by clicking the  button.

The sidebar can be resized by dragging its edge across the workspace.

If the edge is dragged close to the right-hand edge, the sidebar will be hidden. It can be shown again by selecting the 'Show sidebar' option in the View menu.



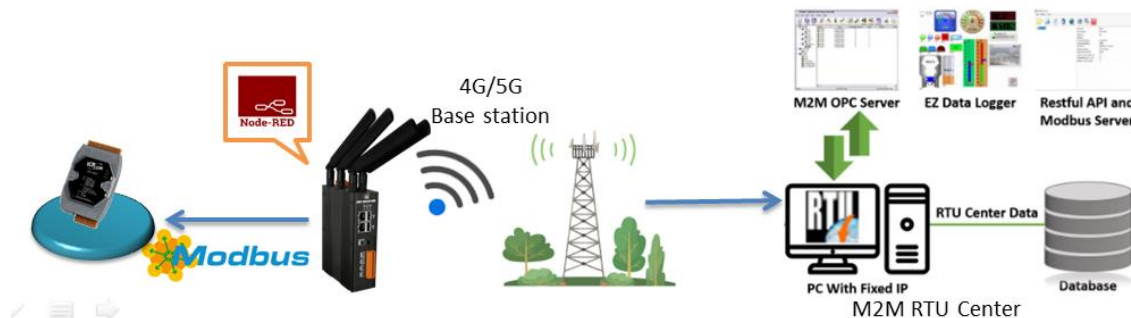
Editor Sidebar

For more information, please refer to the official website: <https://nodered.org/docs/user-guide/>

4. Demo

4.1 Demo1 - RTU IO and GPS positioning

This example shows the steps to get RTU IO and GPS data.



User must enable RTU client function to get and share Modbus IO and GPS data and then Node-Red will display IO and GPS location in web HMI.

(1) Enable RTU Client Function

The screenshot shows the ICP-DAS web interface. The left sidebar has a menu with options like System, VxServer, RTU Client, and SNMP. The 'RTU Client' option is selected. The main content area shows a configuration table with fields for Main Info., Modbus Device, and FTP / Email. The 'Enable Firmware' checkbox is checked.

Main Info.	Modbus Device	FTP / Email
Server Address	192.168.12.2	
Server Port	10000	default=10000
Station ID	1	1~65535
Data Update Period(sec.)	3	0~86400 (0=disable)
Heartbeat Period(sec.)	0	1~86400 (a day)
Baud Rate (for Modbus/RTU)	9600	bps
Data Bit	8	
Parity	N	
Stop Bit	1	
Modbus Timeout (ms)	1000	50~99999, default=1000
Enable Firmware	☒ Enable	
Firmware Version	V1.1.1 2016/10/07	

(2) Select "Custom" in the list, and then press the "Add" button, the web will display all I/O number information, as shown below.

Modify the "Device Name", "Device ID", "IP", "DI/DO/AI/AO Number", "DI/DO/AI/AO Address" of Modbus device, and then click "Modify".

Main Info.		Modbus Device		FTP / Email	
Modbus Device Number : 1				Add Custom ▼	
1	Name : 1_Custom			Edit Delete	
Device Name	1_Custom	Max Length=20			
Device ID	1	1~255			
IP	192.168.255.100	IP/COM1/COM2/COM3 COM is for Modbus/RTU, empty is for COM2			
Port	502	Default=502, 1~65535			
DI Number	4	0~32			
DO Number	4	0~32			
AI Number	4	0~16			
AO Number	4	0~16			
DI Address	0	0~65535			
DO Address	0	0~65535			
AI Address	0	0~65535			
AO Address	0	0~65535			
Modify Cancel					

(3) Select the "Modbus Test" function, and then press the "Test" button to test the settings. If the result is successful, follow the next step.

--Time
--System Service
VxServer
--VxServer
RTU Client
--RTU Client
--FTP Test
--Email Test
--Modbus Test
V1.1.2 B07
2014/07/28

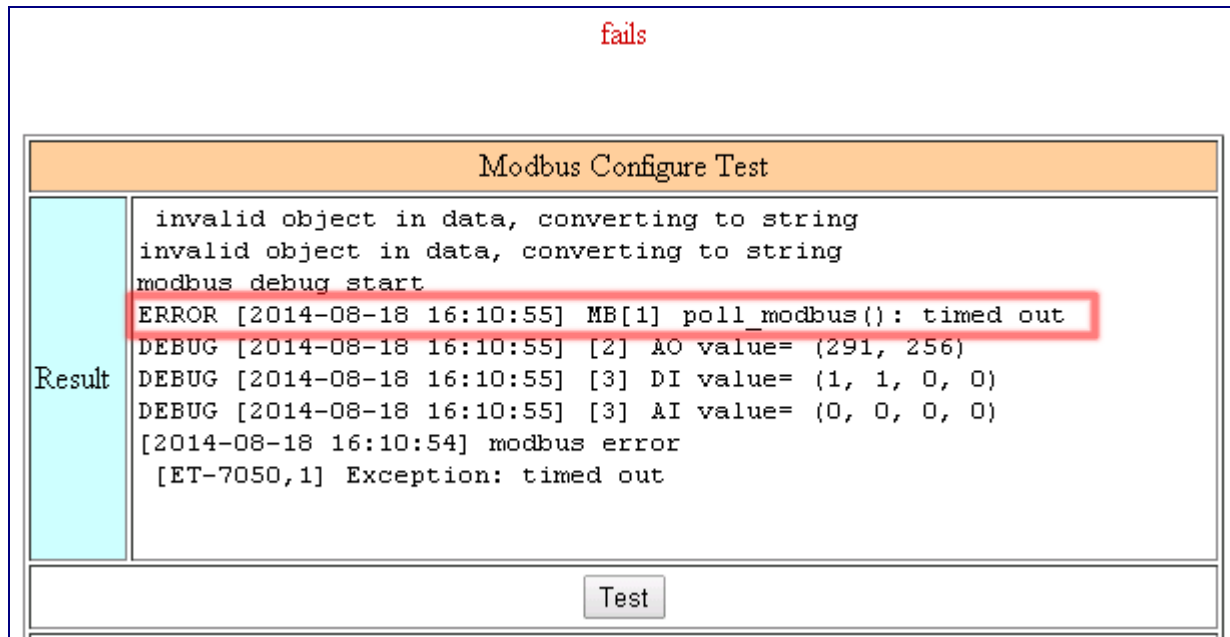
success

Modbus Configure Test

invalid object in data, converting to string
invalid object in data, converting to string
modbus debug start
DEBUG [2014-08-18 15:55:56] [1] DI value= (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)
DEBUG [2014-08-18 15:55:56] [1] DO value= (0, 0, 0, 0, 0, 0)
DEBUG [2014-08-18 15:55:56] [2] AO value= (291, 256)
DEBUG [2014-08-18 15:55:56] [3] DI value= (1, 1, 0, 0)
DEBUG [2014-08-18 15:55:56] [3] AI value= (0, 0, 0, 0)

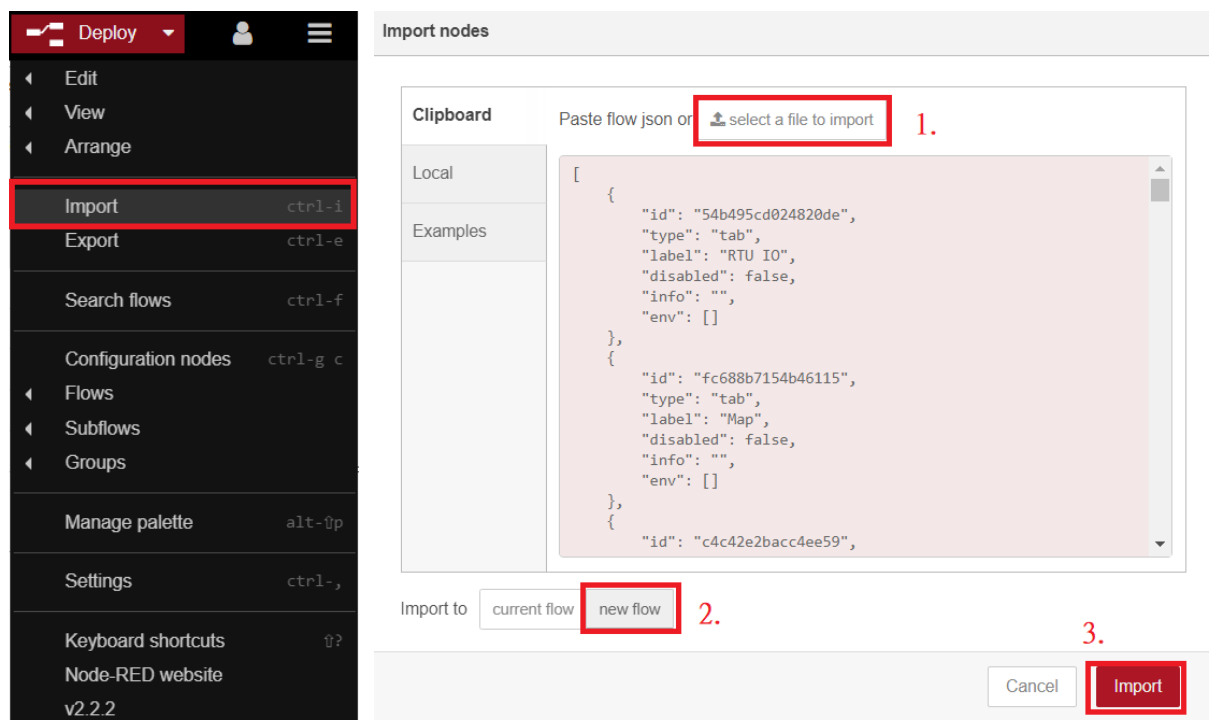
Test

If the result failed, please check your settings or wiring.



(4) Import flow from demo1.json file

use the "Menu -> Import -> select a file to import".



Please refer: 3.2.4 Importing and Exporting Flows for detail.

Warning:

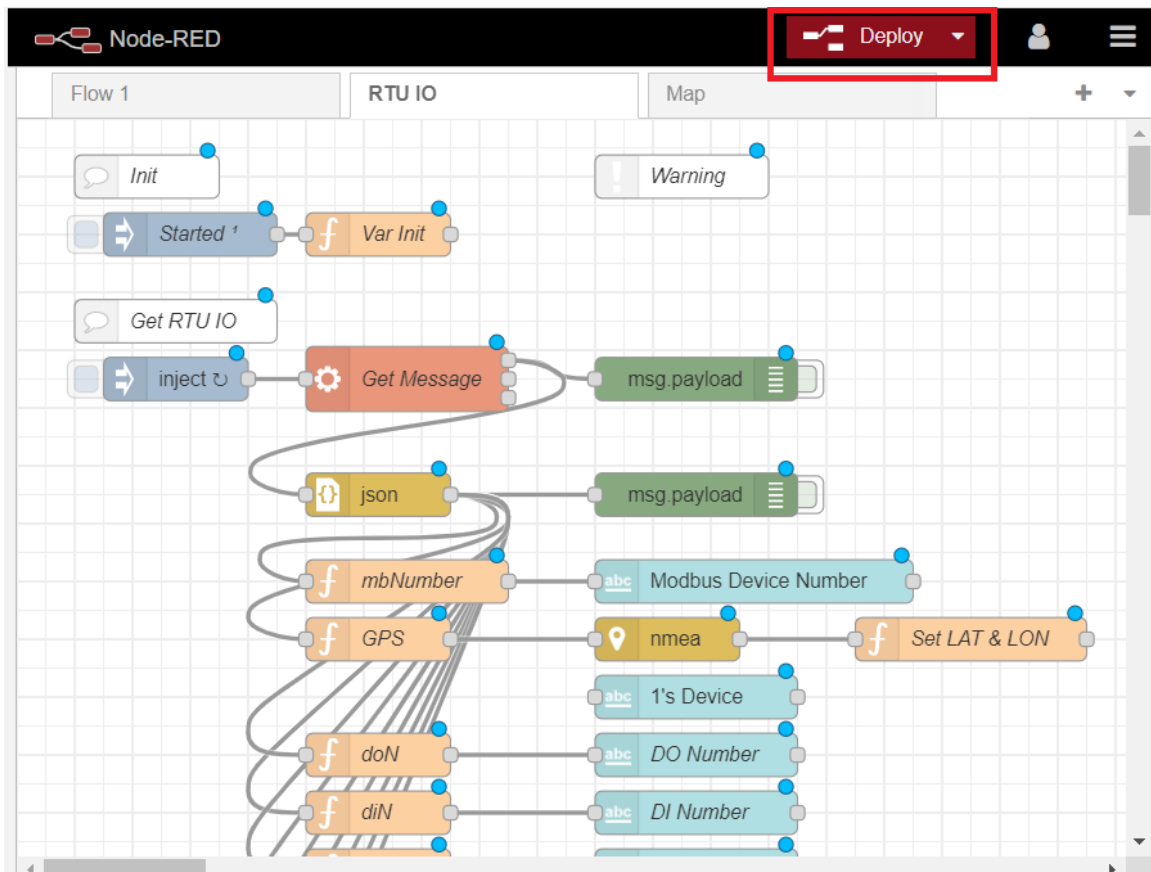
****This demo requires the following modules to be installed **:**

node-red-dashboard
node-red-contrib-nmea
node-red-contrib-ui-led
node-red-contrib-web-worldmap

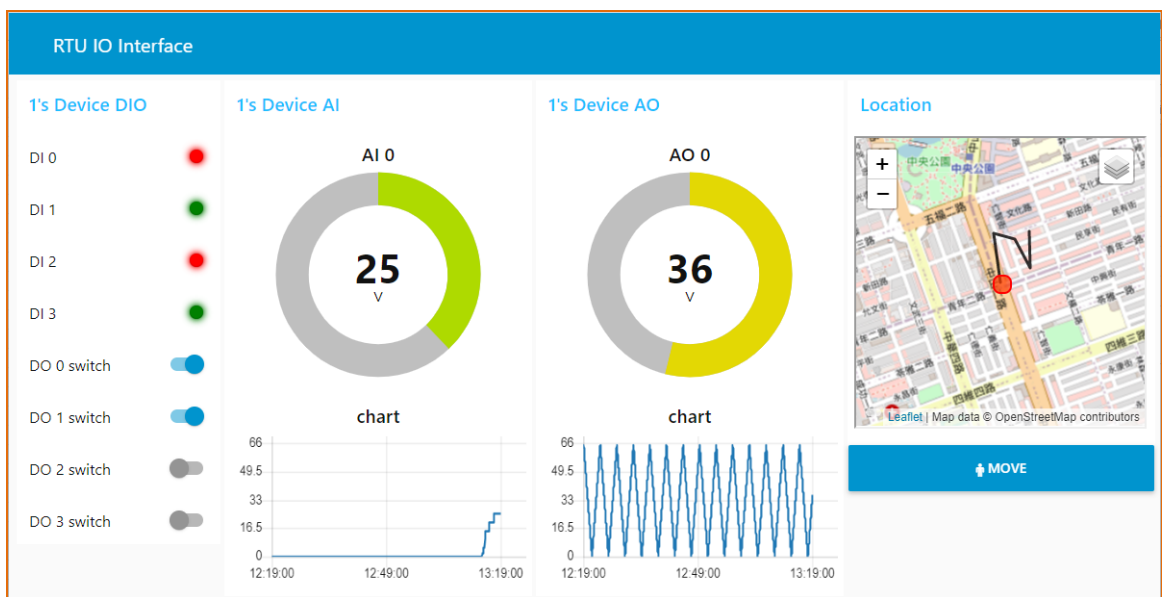
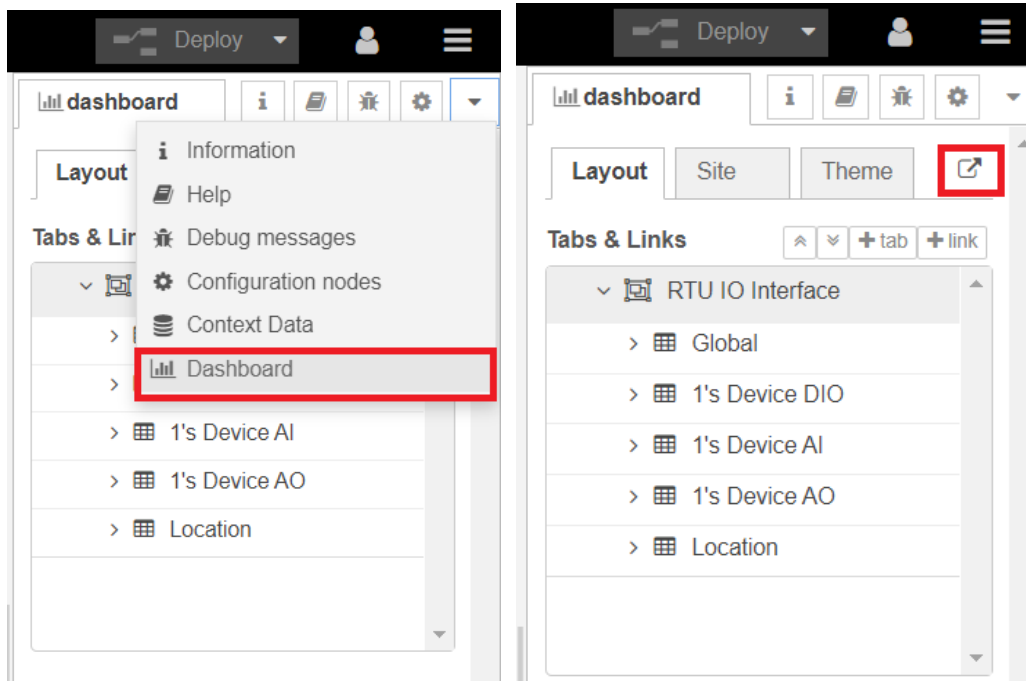
****Install module by:****

use the "Menu -> Manage palette -> Install" option and search for module

(5) Click the deploy button to install the program.

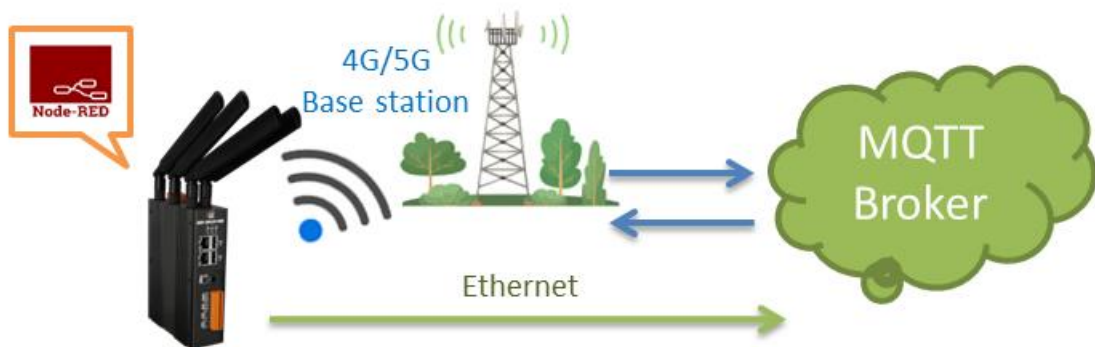


(6) Select dashboard in the sidebar and then click  button to show dashboard



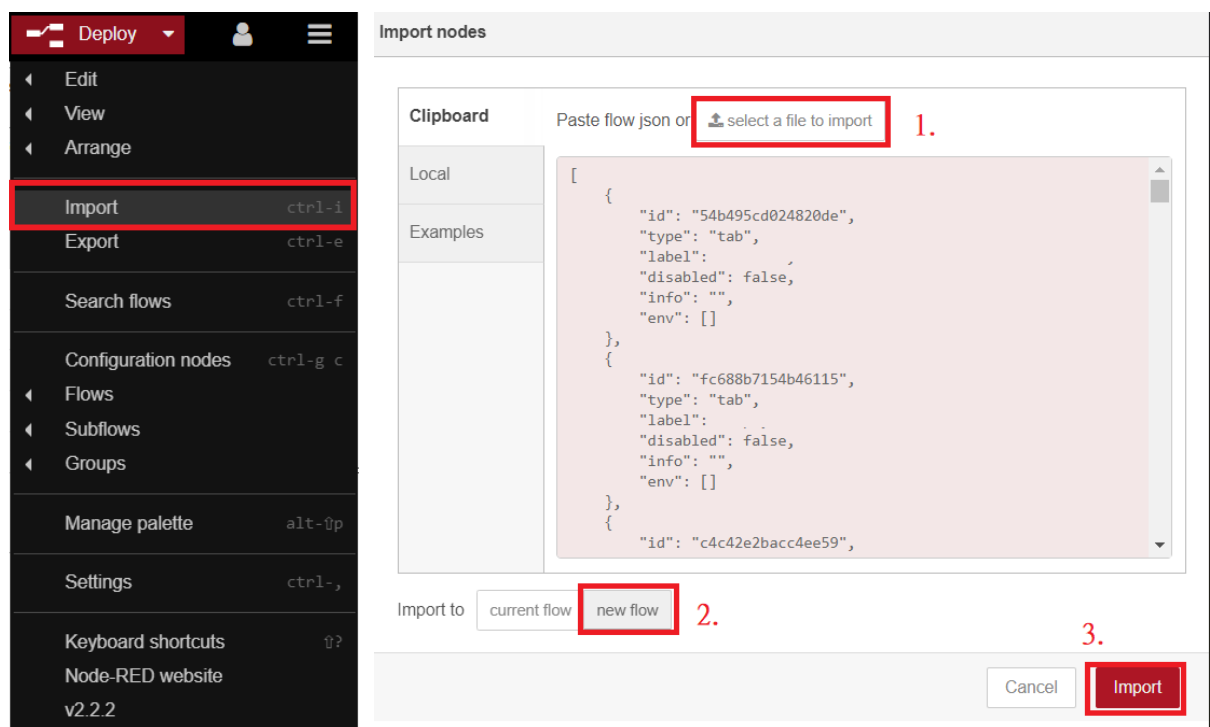
4.2 Demo2 - MQTT publish & subscribe

This example shows the steps to subscribe and publish MQTT messages.



(1) Import flow from demo2.json file

use the "Menu -> Import -> select a file to import".

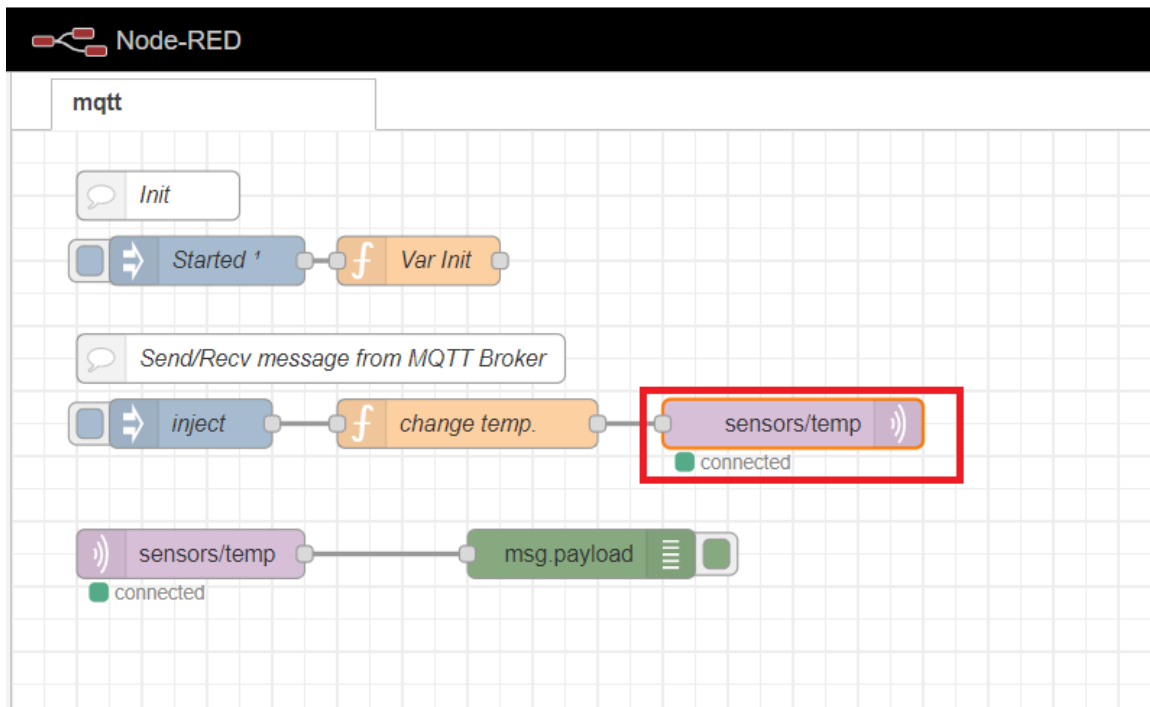


Please refer: 3.2.4 Importing and Exporting Flows for detail.

(2) Click the deploy button to install the program.



(3) Double click mqtt out node (for messages publish) and set server and topic for your mqtt broker.



Edit mqtt out node

Delete Cancel **Done**

Properties

1. Server 122.147.137.139:1883

Topic sensors/temp

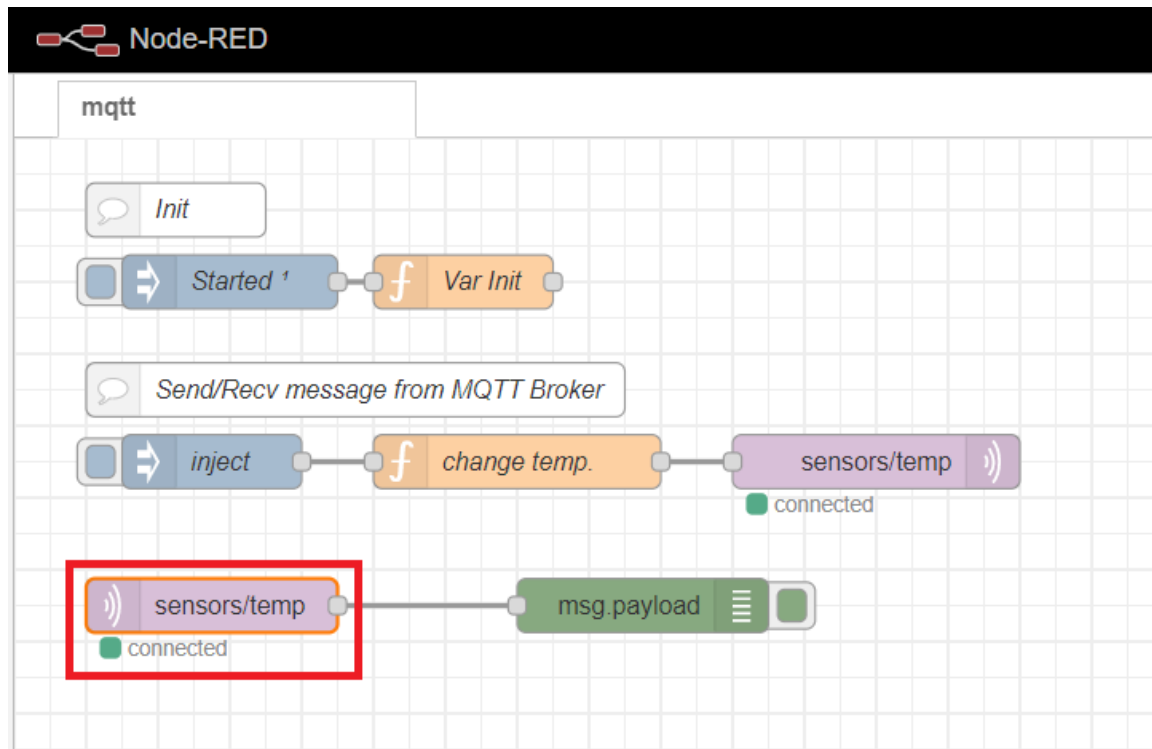
QoS Retain

Name Name

Tip: Leave topic, qos or retain blank if you want to set them via msg properties.

2.

(4) Double click mqtt in node (for messages subscribe) and set server and topic for your mqtt broker.



Edit mqtt in node

Delete Cancel Done

⚙ Properties

🌐 Server 122.147.137.139:1883

Action Subscribe to single topic

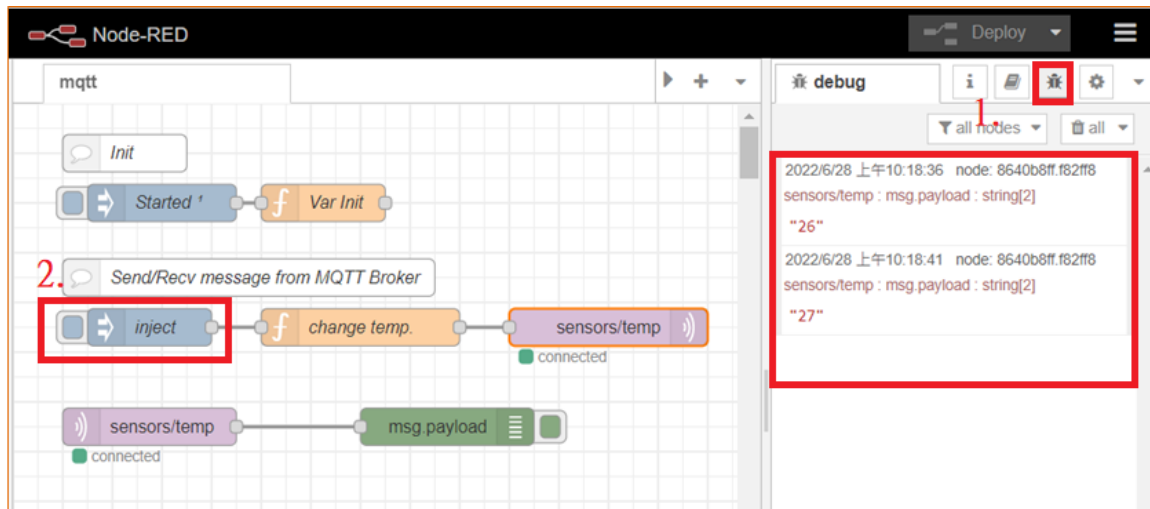
📄 Topic sensors/temp

🌐 QoS 2

👉 Output auto-detect (string or buffer)

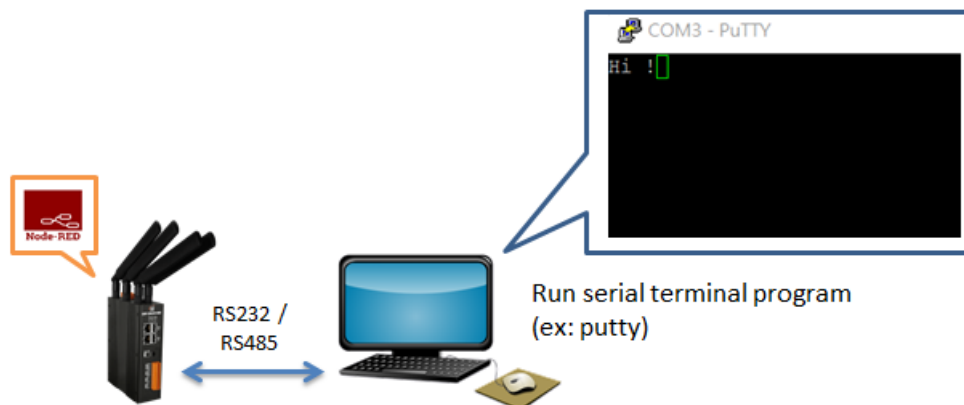
🏷 Name Name

- (5) Select the debug icon to display the debug window and click inject node to publish mqtt message and the subscription message will be displayed in the debug window.



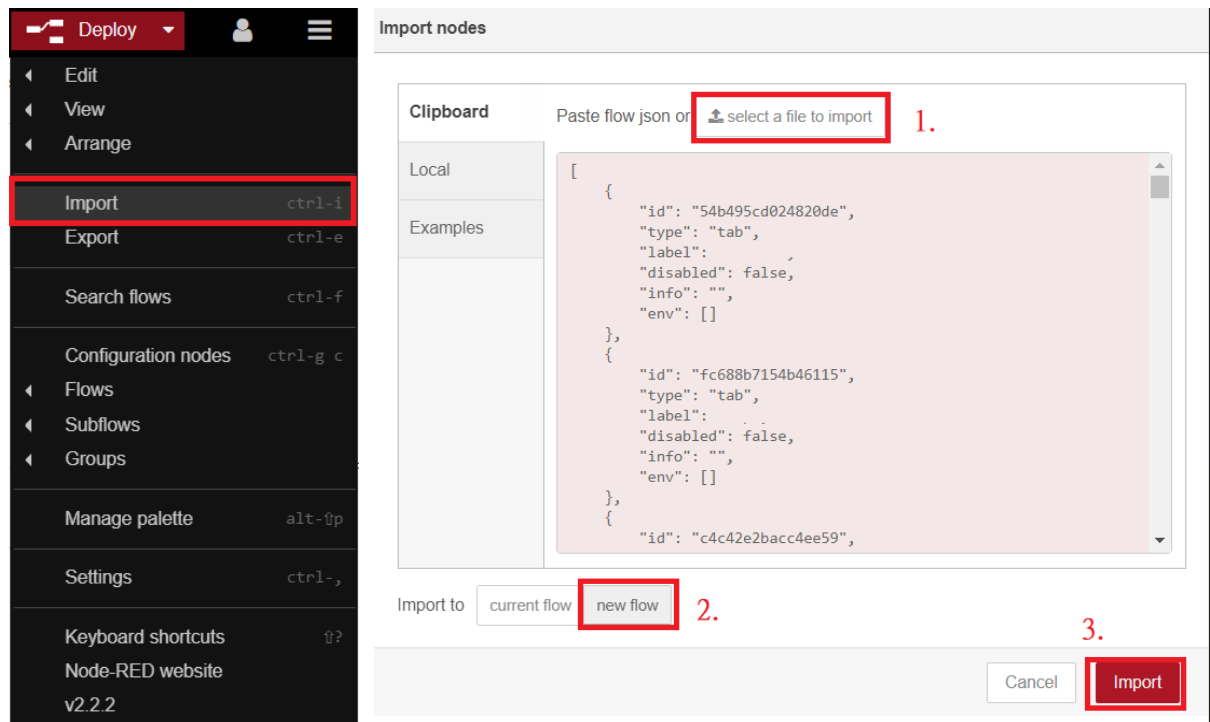
4.3 Demo3 - Read/Write message by serial port

This example shows the steps to read/write messages by serial port.



- (1) Import flow from demo3.json file

use the "Menu -> Import -> select a file to import".



Please refer: 3.2.4 Importing and Exporting Flows for detail.

Warning:

****This demo requires the following modules to be installed **:**

node-red-node-serialport

****Install module by:****

use the "Menu -> Manage palette -> Install" option and search for module

****Device node of COM Port ****

COM1: /dev/ttyUSB0

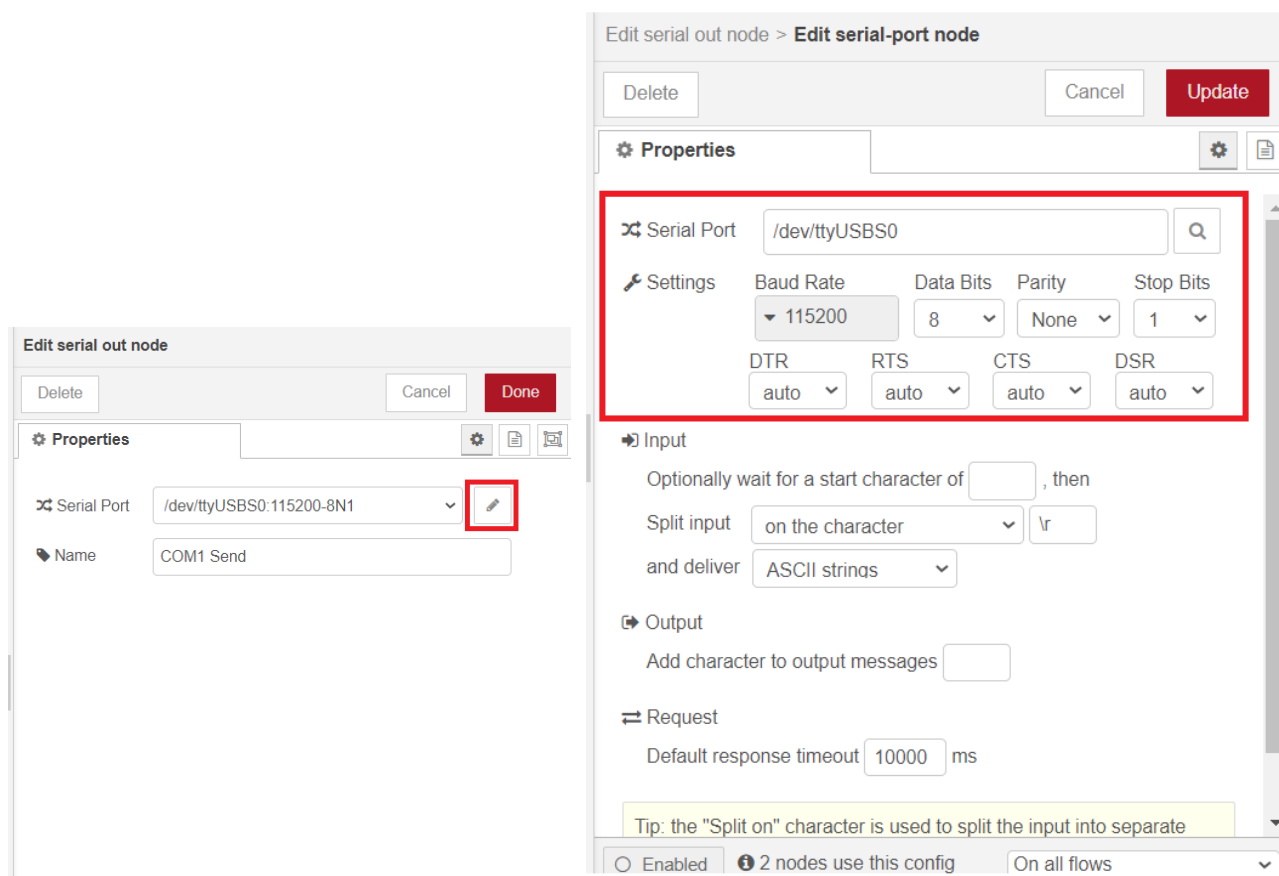
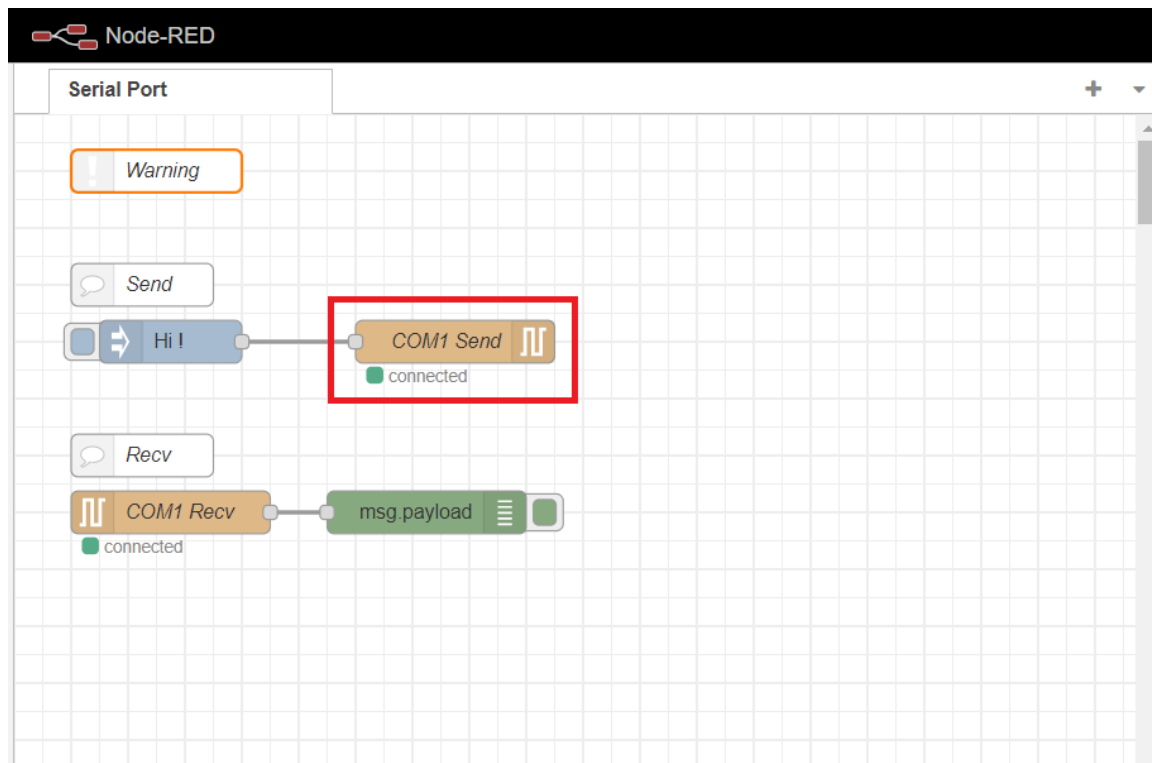
COM2: /dev/ttymxcl

COM3: /dev/ttyUSB2

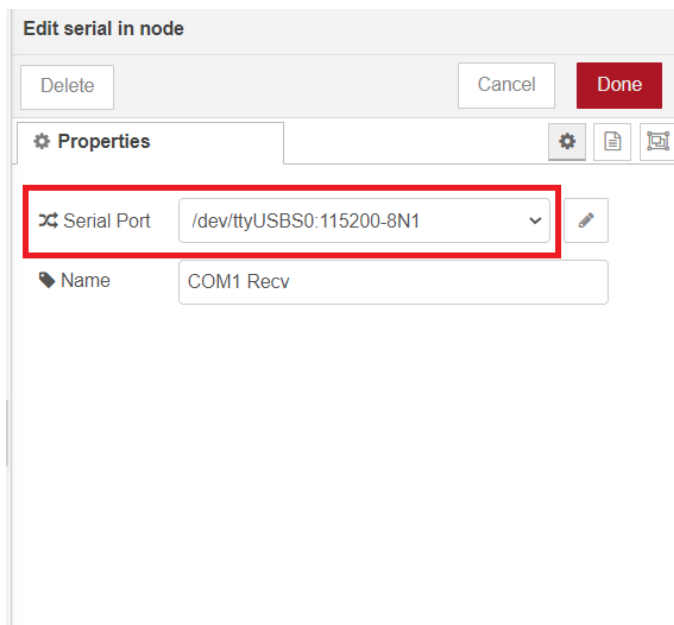
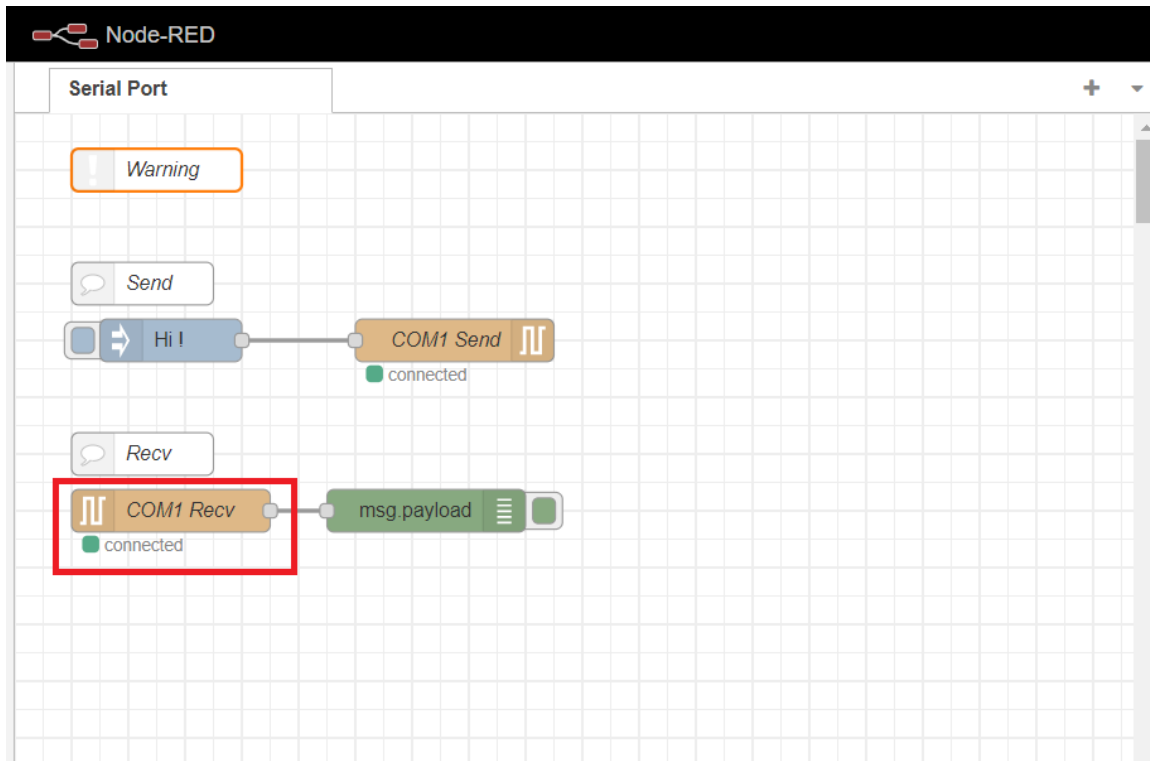
(2) Click the deploy button to install the program.



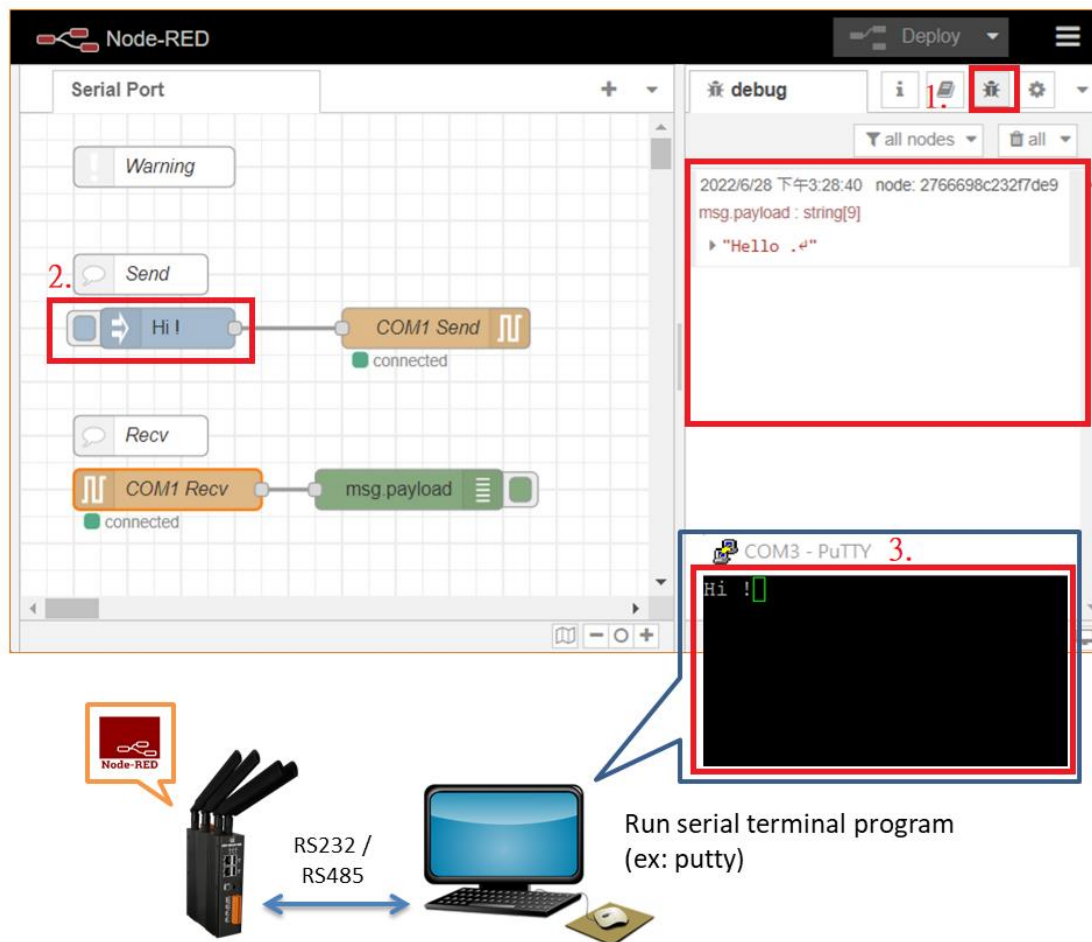
(3) Double click serial out node and set serial port and settings.



(4) Double click serial in node and select serial settings in previous step..



(5) Select the debug icon to display the debug window and click inject node to send string to terminal program and the received string will be displayed in the debug window.



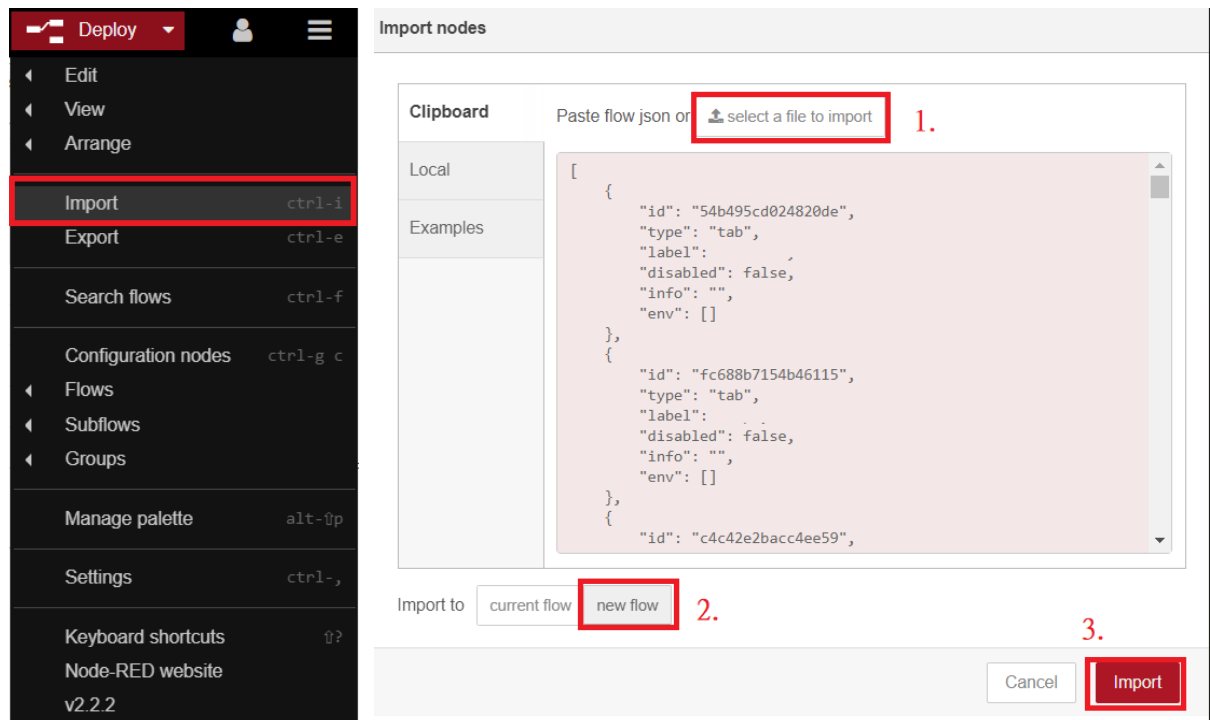
4.4 Demo4 - Capture image via webcam

This example shows the steps to capture image via webcam.



(1) Import flow from demo4.json file

use the "Menu -> Import -> select a file to import".



Please refer: 3.2.4 Importing and Exporting Flows for detail.

Warning:

****This demo requires the following modules to be installed **:**

node-red-contrib-usbcamera
node-red-contrib-image-tools

****Install module by:****

use the "Menu -> Manage palette -> Install" option and search for module

**** Webcam must support UVC driver/standard ****

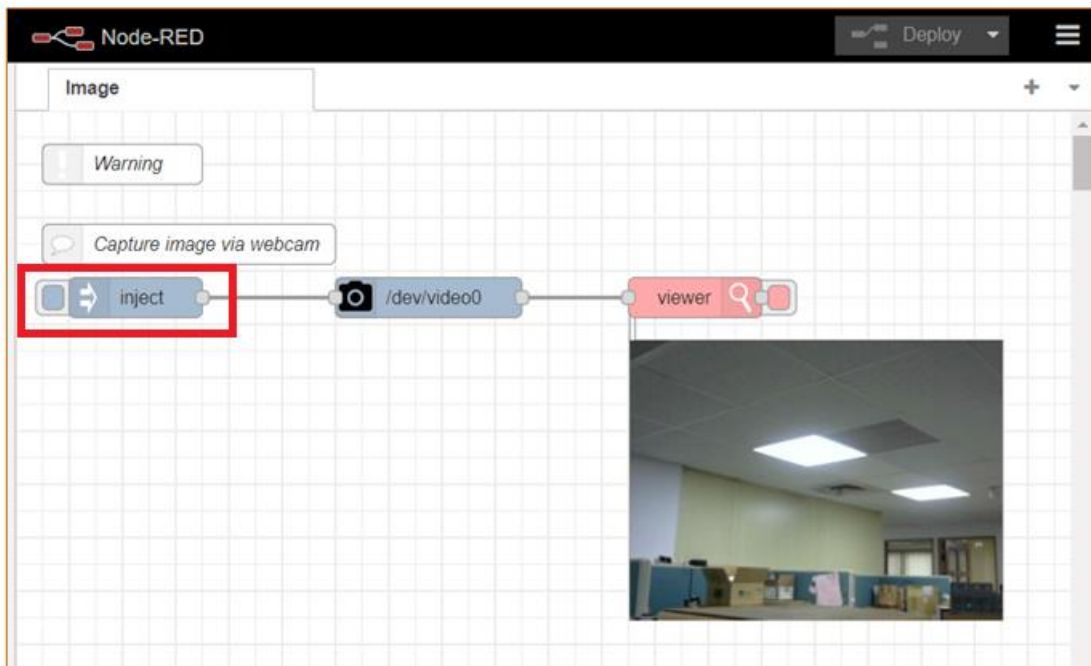
The UVC (also USB video class) is a USB device class that describes devices capable of streaming video like webcams.

See https://en.wikipedia.org/wiki/USB_video_device_class for details.

(2) Click the deploy button to install the program.



(3) Click on inject node to capture image



4.5 Demo5 - Machine Learning with Google's Teachable Machine

This example shows the steps to capture image via webcam and classify images with machine learning models.

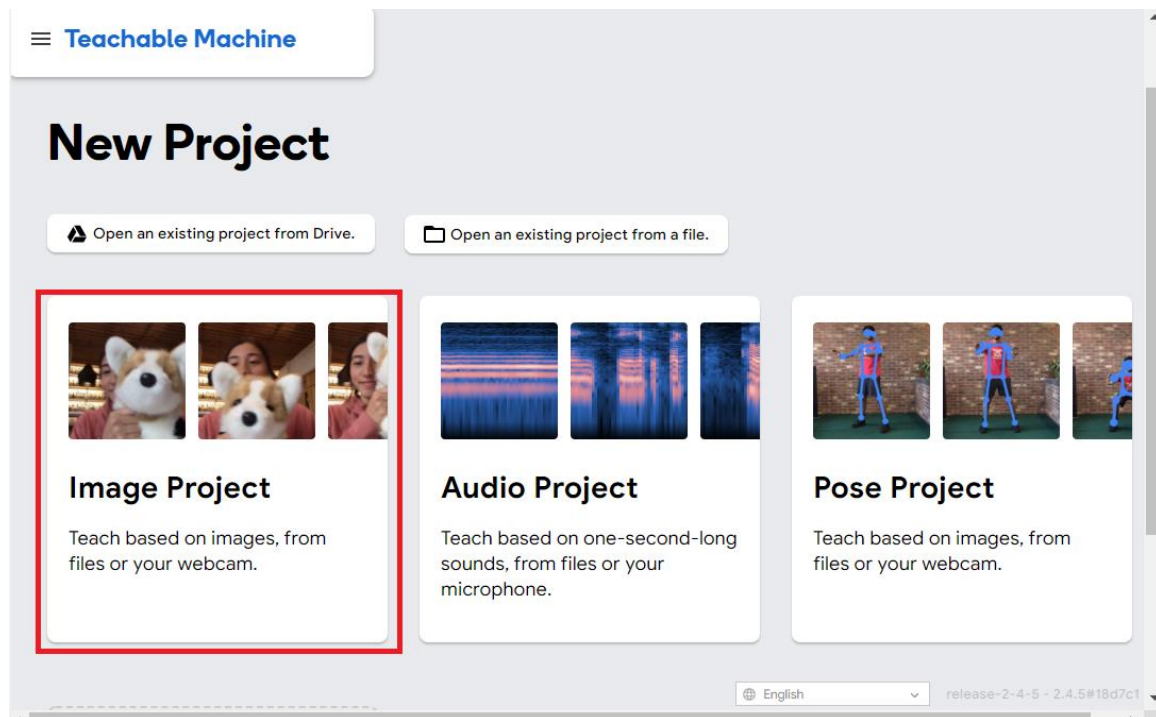


Teachable Machine is Google's AutoML platform which is a web-based tool that makes

creating machine learning models fast, easy, and accessible to everyone.

See <https://teachablemachine.withgoogle.com/> for details.

(1) Click "Image Project" on the Teachable Machine platform to create a new project

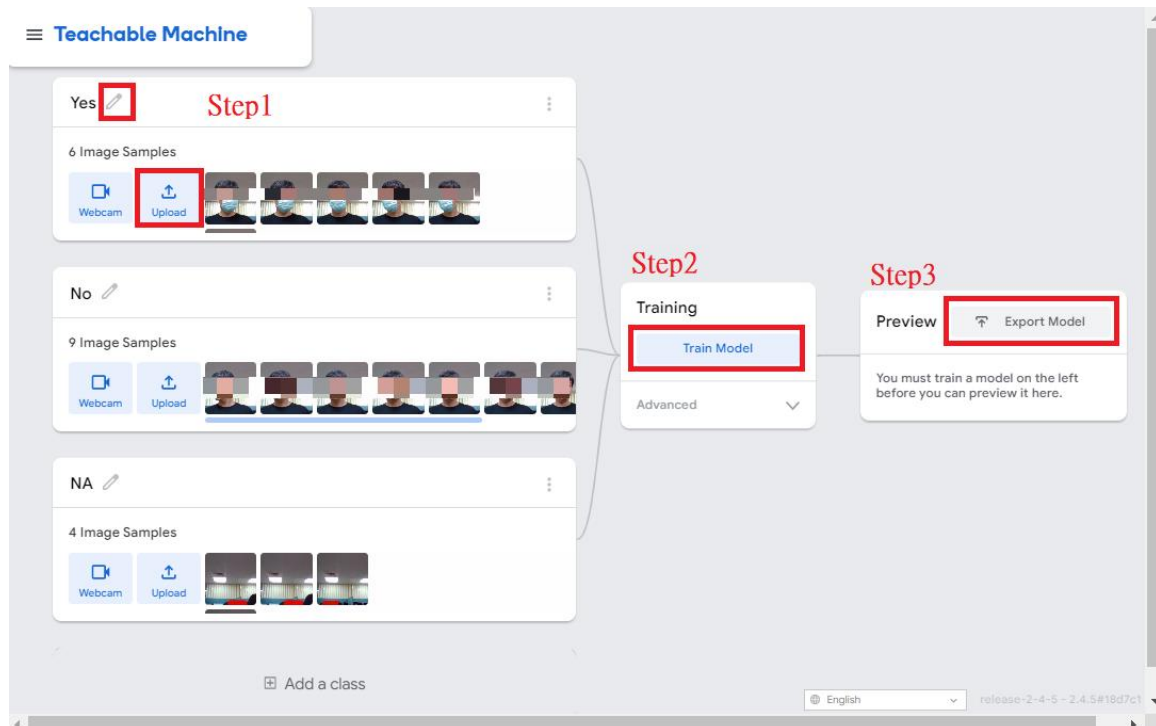


(2) Create and Upload Model

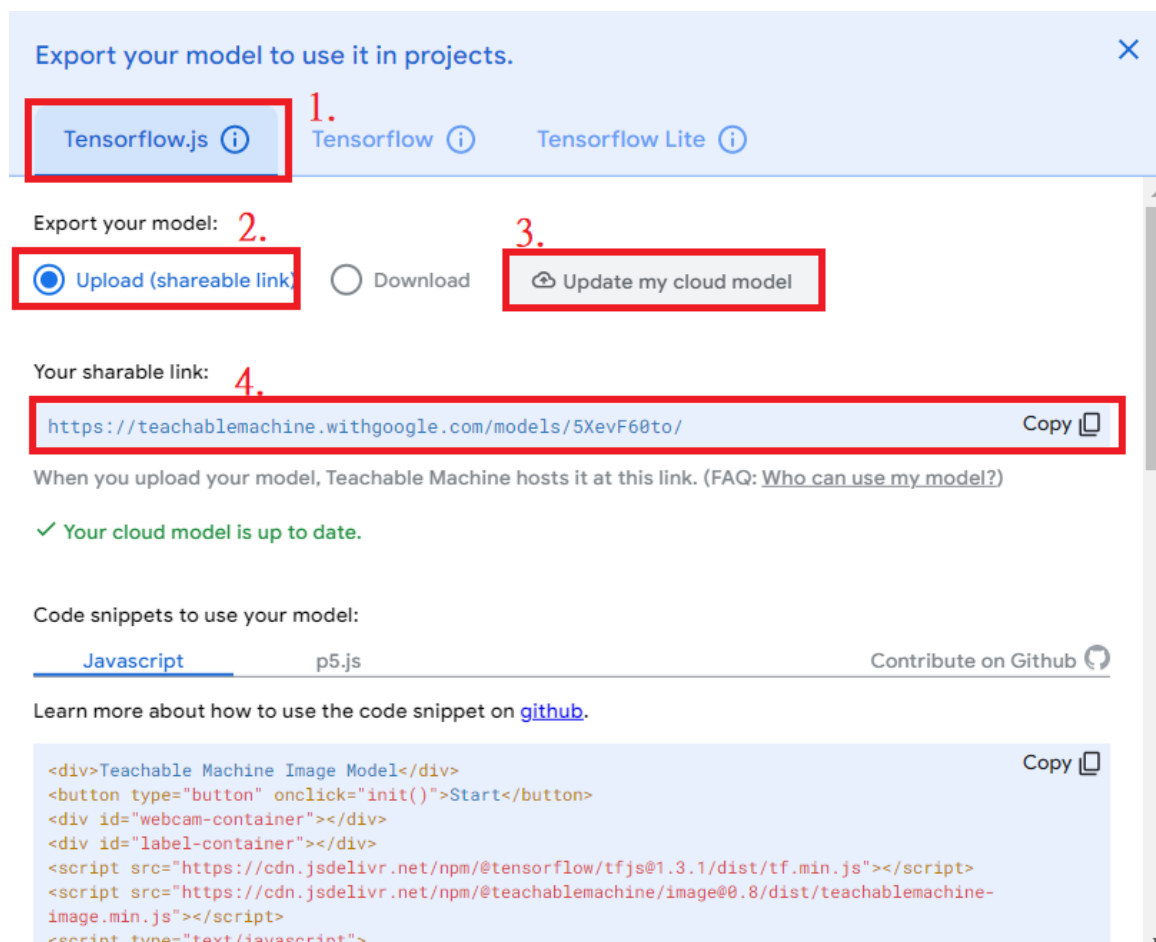
Step1: upload your image

Step2: training model

Step3: Export Model

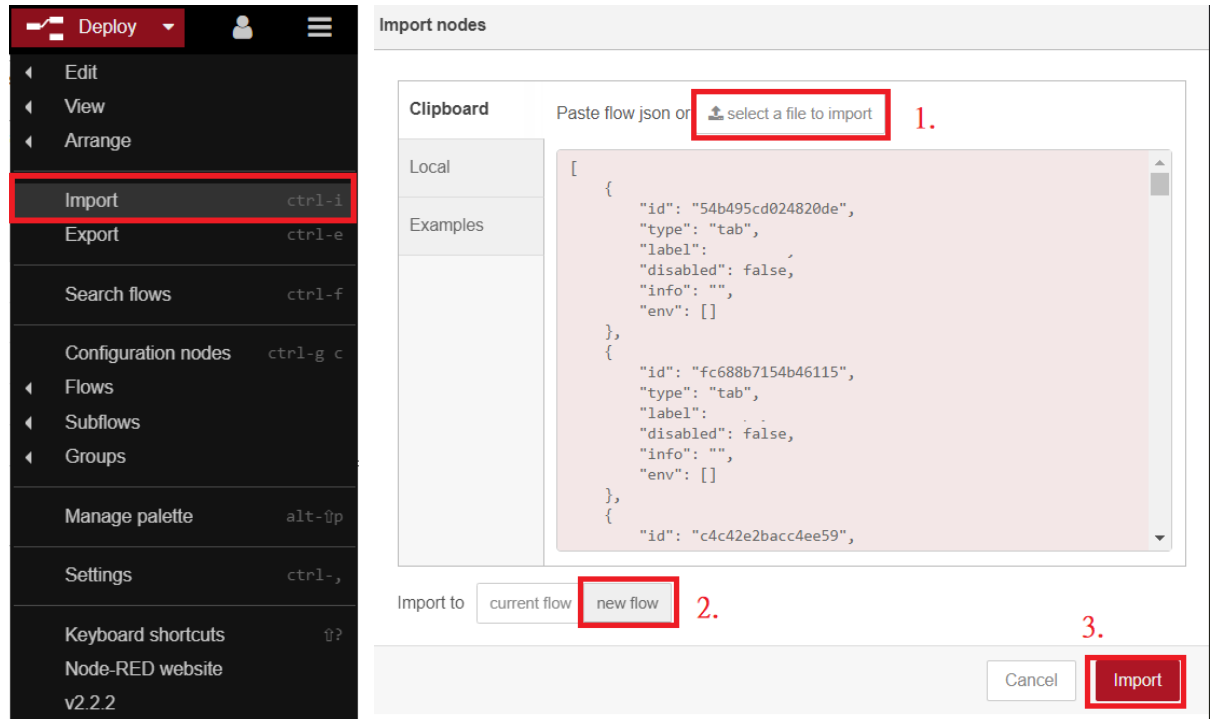


Step4: Upload model and copy your sharable link.



(3) Import flow from demo5.json file

use the "Menu -> Import -> select a file to import".



Please refer: 3.2.4 Importing and Exporting Flows for detail.

Warning:

****This demo requires the following modules to be installed **:**

node-red-contrib-usbcamera
node-red-contrib-image-tools
node-red-contrib-teachable-machine

****Install module by:****

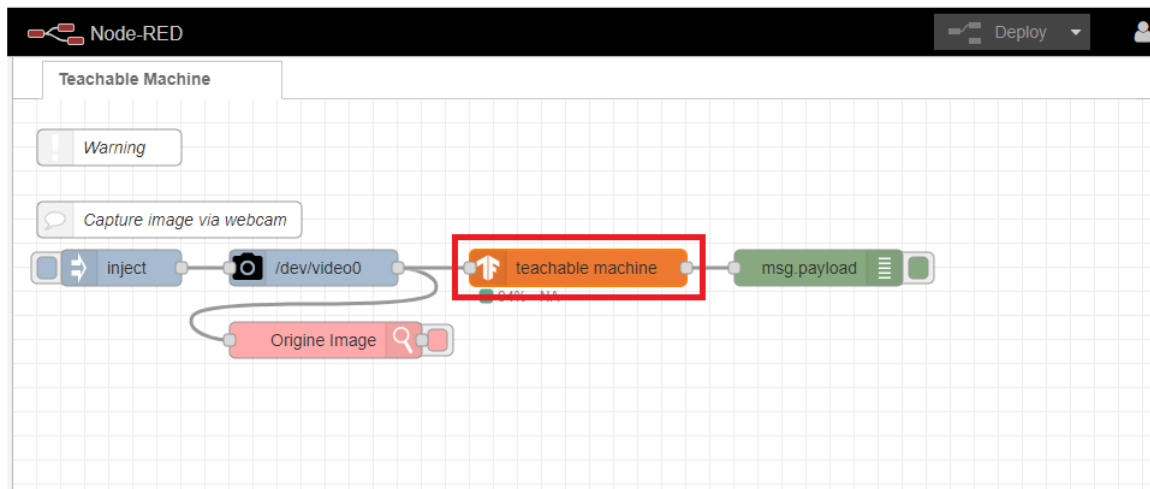
use the "Menu -> Manage palette -> Install" option and search for module

**** Webcam must support UVC driver/standard ****

The UVC (also USB video class) is a USB device class that describes devices capable of streaming video like webcams.

See https://en.wikipedia.org/wiki/USB_video_device_class for details.

(4) Edit teachable machine node, paste your shareable link into the Url textbox.

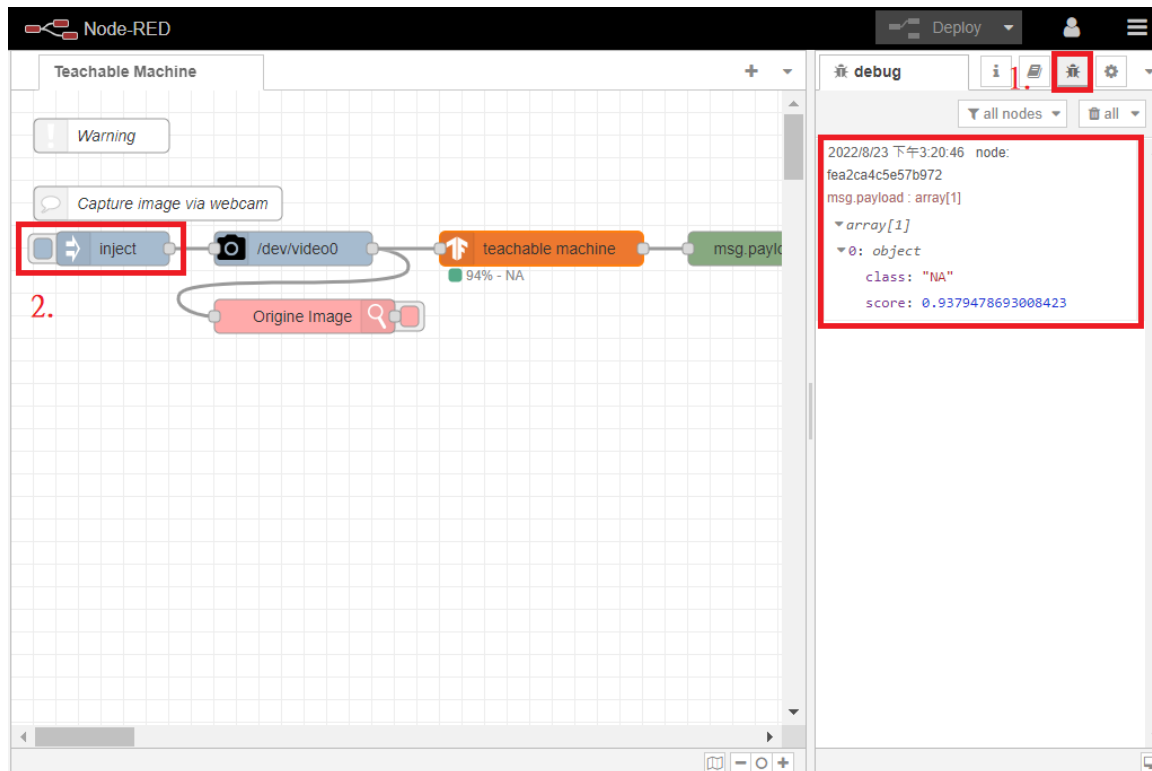


The screenshot shows the 'Edit teachable machine node' dialog box. It has a 'Delete' button, 'Cancel' and 'Done' buttons. Under the 'Properties' tab, the 'Mode' is set to 'Online'. The 'Url' field is highlighted with a red rectangular box and contains the text 'https://teachablemachine.withgoogle.com/models'. Other fields include 'Output' set to 'Best prediction', an unchecked checkbox for 'save original image in msg.image.', and a 'Name' field with the text 'Name'. At the bottom, there is an 'Enabled' checkbox.

(5) Click the deploy button to install the program.



- (6) Select the debug icon to display the debug window, click the inject node to capture the image, and the recognition result will be displayed in the debug window.



Appendix A. Revision History

This chapter provides revision history information to this document.

The table below shows the revision history.

Version	Date	Description of changes
1.0.0	2022-08-10	The First Release Revision